

# 인공지능 프로그래머는 인간 프로그래머를 능가할 수 있는가?

박해준

2022.11.11.

## Abstract

최근 몇 년 사이 인공지능 프로그래밍 기술이 발달하며 인간 프로그래머와의 비교가 불가피해졌다. 인간 프로그래머는 인공지능 프로그래머에 비해 결과물의 질에서 앞서고, 과정(소모 자원)에서 뒤떨어진다. 프로그래머의 본질에 충실하며 적절히 인공지능을 이용한다면 프로그래머로서의 경쟁력을 갖출 수 있을 것이다.

깃허브(GitHub)의 코파일럿(Copilot)에 대해 들어 보았는가? 이는 GPT-3 모델을 채택한 인공지능으로서, 코드를 자동으로 완성시켜준다. 예를 들어, `int sum(a, b)`라는 함수 이름과 더불어 적절한 주석을 제공하면 `return a+b`라는 문장을 작성한다. 이는 프로그램 작성 시간을 크게 줄일 수 있는 혁신적인 기술임에 틀림없다. 코드를 작성하는 시간을 줄임은 물론, 인간의 자잘한 실수로부터 비롯되는 큰 디버깅의 여지를 없앨 수 있기 때문이다. 그렇다면 과연 이 기술이 전 세계의 프로그래머를 위협할 수 있을까? 인공지능이 자신을 낳은 프로그래머를 뛰어넘은 더 나은 프로그래머가 될 수 있을까?

논의를 시작하기 전, 문제를 정확히 정의할 필요가 있다. ‘인공지능’이 프로그래머를 능가한다<sup>1</sup>라고 함은 프로그래머가 하는 일을 인공지능이 더 잘 해냄을 뜻한다. 프로그래머는 클라이언트의 요구 혹은 지시를 실제로 수행하는 프로그램을 프로그래밍 언어를 통하여 작성한다. 우리가 논하고자 하는 바는 클라이언트가 인간이 아닌 ‘인공지능 프로그래머’에게 ”이러한 일을 하는 프로그램을 만들어 주세요”라고 지시했을 때, 해당 인공지능 프로그래머가 인간보다 여러 방면에서 더 나은지로 치환될 수 있을 것이다.

프로그래머 A가 프로그래머 B보다 나은지를 판단하는 척도에는 여러 가지가 있지만, 나는 두 가지의 분류를 제안하고 싶다. 첫 번째는 결과물의 질이다. A가 B보다 버그가 적고 속도가 빠르며, 그러면서도 의도에 잘 부합하는 프로그램을 작성한다면 A가 B보다 나은 프로그래머라고 마땅히 부를 수 있다. 둘째는 과정의 차이이다. A가 B보다 프로그램을 완성하기까지 걸리는 자원이 적다면 역시 A가 B보다 나은 프로그래머라고 할 수 있다. (문제를 간단히 하기 위해 ‘자원’을 돈과 시간으로 생각하자.)

정의된 문제를 상상으로 옮기자. 당신은 프로그래밍을 전혀 할 줄 모른다. 당신은 두 수를 입력받아 합을 출력하는 프로그램을 만들고자 하고, 이를 위해 프로그래머를 고용하려 한다. 주어진 선택지는 다음의 두 가지이다. (논하지 않은 다른 조건은 같다고 하자.)

- A: 완성까지 한 달이 걸리지만 버그의 가능성을 잘 틀어막은 프로그램을 작성. 만 원 요구.
- B: 완성까지 하루가 걸리지만 버그의 가능성을 활짝 열어놓은 프로그램을 작성. 백 원 요구.

당신이라면 어떤 프로그래머를 선택하겠는가? 사람마다 중시하는 가치가 다르기 때문에 이에 대한 통일된 답을 얻기는 어려울 것이다. 그렇지만 대부분의 경우 A보다는 B가 인공지능에 가까운 프로그래머임에 동의할 터이다. 속도에 있어서는 인간이 인공지능을 이길 수 없음이 이미 여러 사례에서 증명되어 왔기 때문이다. 비용 역시 한 번 사면 두고두고 쓸 수 있는 ‘프로그래머 프로그램’이 인간 프로그래머보다 훨씬 효율적일 것이다.

다시 프로그래머 A와 B를 비교하는 두 가지의 척도 분류로 돌아가자. ‘과정’, 즉 비용과 시간 면에서는 인공지능 프로그래머가 인간 프로그래머를 압도함이 자명하다. 인간 프로그래머에게 남은 승부처는 자연스레 ‘결과물의 질’이 된다. 이에 속하는 척도에는 다음이 있다. (이하의 용어는 이후의 논의를 편하게 하기 위해 임시로 붙인 이름이다.)

- 정확성: 작성한 프로그램이 클라이언트의 의도에 잘 부합하는가?
- 무결성: 작성한 프로그램이 오류를 만들어내지 않는가 (오류를 방지하는가)?

<sup>1</sup>여기서 인공지능은 기존에 작성되어 있던 프로그램을 학습하여 주어진 지시를 수행하는 새로운 프로그램을 만드는 소프트웨어를 의미한다. 프로그램을 학습하는 프로그램에 관한 재미있는 이야기거리와 논란이 많지만 주제와 맞지 않기에 생략한다.

- 효율성: 작성한 프로그램이 시공간적 자원을 덜 소모하는가?

정확성은 클라이언트의 요구, 즉 자연어를 인공지능 프로그래머가 이해할 수 있는 구체적 사양(specification)으로 옮기는 과정을 수반한다. 프로그래밍을 전혀 모르는 당신 입장에서는 자연어를 사양으로 옮기는 또 다른 인공지능을 고용할 수밖에 없고, 이는 불필요한 낭비를 뜻한다. 또한 인간 프로그래머는 클라이언트에서 제시하지 않은 몇 가지 사양을 능동적으로 덧붙일 수 있다. 적어도 정확성 면에서는 인간이 인공지능 프로그래머를 앞서는 바이다.

무결성과 효율성은 논의하기 힘든 부분이 많다. 인공지능 프로그래머가 얼마나, 어느 수준의 프로그램을 학습하였는지에 따라 내놓는 결과물이 크게 달라지기 때문이다. 그러므로 이 둘에 대한 논의는 차치하도록 하자.

정리하면, 인간 프로그래머는 인공지능 프로그래머보다 결과물의 질에서 앞서고, 과정에서 뒤떨어진다. 가지고 있는 경쟁력이 다르기에 한쪽이 다른 쪽을 완전히 뛰어넘는다고 선언하기는 힘들어 보인다. 그러나 기술이 발전함에 따라 인공지능이 결과물의 질에서 보이던 약세가 줄어드는 점 역시 엄연한 사실이다.

이러한 상황에서 프로그래머가 갖추어야 하는 덕목은 무엇일까? 다소 과감한 주장일 수 있지만, 나는 인공지능이 잘 할 수 있는 분야를 인공지능으로 넘기는 것이 옳다고 생각한다. 사람이  $\text{sum}(a, b)=a+b$ 와 같은 (비교적) 간단한 프로그램을 작성하는 것은 낭비이고 예상치 못한 오류를 야기한다. 인공지능이 잘할 수 있는 분야는 인공지능에게 맡기고, 사람이 잘할 수 있는 분야는 사람이 도맡아 더 높은 질의 결과물을 내놓는 것이 좋다고 생각한다. 그러기 위해서는 여전히 논리 및 기능의 오브제를 구조화하여 연결하는 방법을 이해하여야 한다. 프로그래밍의 본질에 충실하며 적절히 인공지능을 이용한다면 우수한 프로그래머로서의 경쟁력을 갖출 수 있으리라 생각한다.